HW2 – due Oct 19th

PART I. The Implementation Assignment (team of up to 2 persons)

In this assignment, you will implement the top-down greedy induction algorithm for learning decision trees. In particular, given a set of training examples you implementation should learn: (1) a decision stump, i.e., a decision tree with only one test; and (2) a depth-2 decision tree, that is there are at most two tests from the root to each leaf. Use information gain as the selection criterion for building the decision tree. Test your implementation on the monk data set that is provided to you.

The monk dataset is a synthetically generated data set that was created for a comparison study on different machine learning algorithms. For those of you who are interested, this data set (and two other related data sets), and the comparison results are described in the following paper:
The MONK's Problems - A Performance Comparison of Different Learning algorithms by a group of machine learning researchers.

The particular data set provided for this implementation assignment is the monk 1 data set. There are 6 features, the meanings of which are provided below:

\[
\begin{align*}
x_1: & \text{ head_shape } \in \text{ round, square, octagon} \\
x_2: & \text{ body_shape } \in \text{ round, square, octagon} \\
x_3: & \text{ is_smiling } \in \text{ yes, no} \\
x_4: & \text{ holding } \in \text{ sword, balloon, flag} \\
x_5: & \text{ jacket_color } \in \text{ red, yellow, green, blue} \\
x_6: & \text{ has_tie } \in \text{ yes, no}
\end{align*}
\]

Note that the provided data use different numbers to represent the different feature values. For example, feature 1 in the data has three different values 1, 2, and 3, representing 'round', 'square' and 'octagon' respectively. For this assignment, please do not worry about the bias for preferring multi-nominal features and just use the simple information gain (instead of the normalized gain ratio) to evaluate all features.

The class label is a concept defined by the following rule:

(\text{head\_shape} = \text{body\_shape}) \text{ or } (\text{jacket\_color} = \text{red})

That is, if \(x_2=x_1\) or \(x_5=1\), \(y=1\). There are 432 possible examples in total, and 124 were randomly chosen as training examples.

As mentioned, please learn 1) a decision stump; and 2) a depth-2 decision tree from the training data.

In your report, please provide
1) both the learned stump and the depth-2 decision tree. To help grading easier, please provide for each selected test its information gain.
2) The training and testing error rate of the learned stump and depth-2 decision tree.

Finally, please answer the following questions:

Given the formula for generating the class label, please provide a (as compact as possible) decision tree that will correctly classify all training examples. Do you expect
the top-down greedy induction algorithm to learn this optimal tree? Note that your answer should be general, not depending on the specific training set you use in this assignment.

**PART II (Individual assignment)**

1. We have two identical bags. Bag A contains 4 red marbles and 6 black marbles and bag B contains 6 red marbles and 6 black marbles. Now we randomly chose a bag and drew a marble from the chosen bag and it turns out to be black. What is the probability that the chosen bag is bag A?

2. Suppose we have class variable $y$ and three attributes $x_1, x_2, x_3$ and we wish to calculate $P(y = 1 \mid x_1 = u_1, x_2 = u_2, x_3 = u_3)$, and we have no conditional independence information.

   (a) Which of the following sets of probabilities are sufficient for calculation?

   i. $P(y = 1); P(x_1 = u_1 \mid y = 1); P(x_2 = u_2 \mid y = 1); P(x_3 = u_3 \mid y = 1)$

   ii. $P(x_1 = u_1, x_2 = u_2, x_3 = u_3); P(y = 1); P(x_1 = u_1, x_2 = u_2, x_3 = u_3 \mid y = 1)$

   iii. $P(y = 1); P(y = 1 \mid x_1 = u_1); P(y = 1 \mid x_2 = u_2); P(y = 1 \mid x_3 = u_3)$

   (b) Now suppose we know that the variables $X_1, X_2, X_3$ are conditionally independent given the class variable $Y$. Which of the above 3 sets are sufficient now?

3. (Naïve Bayes Classifier) We will use the following training set to build a Naïve Bayes classifier. A, B, and C are three binary attributes and Y is the target class label.

   ![Training Set Table]

   a. Based on the training data, calculate the prior distribution for $Y$: $P(Y)$, with and without Laplace smoothing.

   b. Based on the training data, calculate the distributions $P(A \mid Y), P(B \mid Y)$ and $P(C \mid Y)$, with and without Laplace smoothing.

   c. What prediction will the Naïve Bayes classifier make for a new example $(A=1, B=0, C=0)$, with and without Laplace smoothing?
4. Decision tree learning
Given the following data set:

<table>
<thead>
<tr>
<th></th>
<th>V</th>
<th>W</th>
<th>X</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

The task is to build a decision tree for classifying Y.
(a) Compute the information gain of attributes X, V and W respectively.
(b) Use information gain for selecting test and produce the full decision tree
generated by the top-down greedy algorithm described in class. (Stopping
criterion: stop if all remaining instances belong to the same class.)
(c) Considering the following two strategies for avoid over-fitting.
i. The first strategy stops growing the tree when the information gain of the
   best test is less than a given threshold $\varepsilon$.
ii. The second strategy grows the full tree first and then prunes the tree
    bottom up: start from the lowest level of the tree and prune a sub-tree if
    the information gain of the test is less than a given threshold $\varepsilon$. (Note that
    you should stop checking level $t$ if none of sub-trees at level $t+1$ satisfies
    the pruning criterion.
Let $\varepsilon$ be 0.001 for both cases, write down the resulting tree for each strategy and
compare their training errors.
(d) Discuss the advantages and disadvantages of the two strategies.

5. Show that logistic regression learns a linear decision boundary.